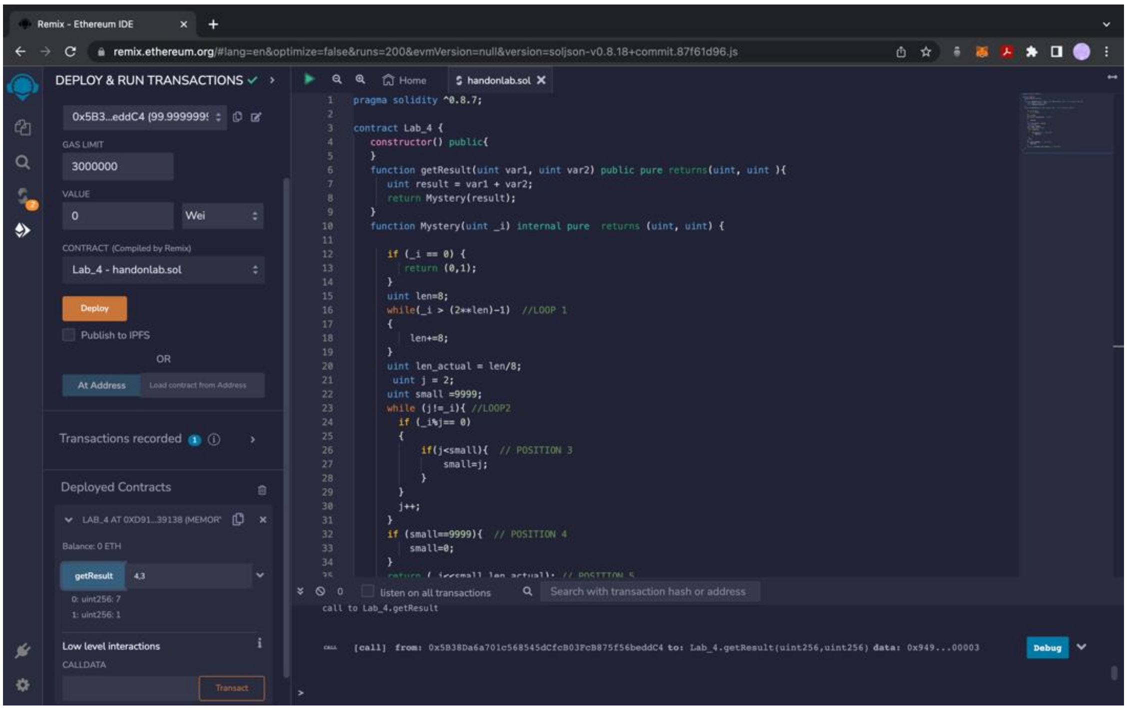
BLOCKCHAIN (UE20CS335)

Name: Vanshika Goel
SRN: PES1UG20CS484
Hands On 4 and 5

Task 2: Understanding the code flow

I. Execute the following program and attach screenshots of the output generated for the following test cases:

a. 4 and 3



DEPLOY & RUN TRANSACTIONS

0x5B3...eddC4 (99.999999

GAS LIMIT: 3000000

VALUE: 0 Wei

CONTRACT (Compiled by Remix): Lab_4 - handonlab.sol

Deploy

□ Publish to IPFS

OR

At Address | Load contract from Address

Transactions recorded 1

Deployed Contracts

LAB_4 AT 0XD91...39138 (MEMOR'

Balance: 0 ETH

getResult 4,3

0: uint256: 7

1: uint256: 1

Low level interactions

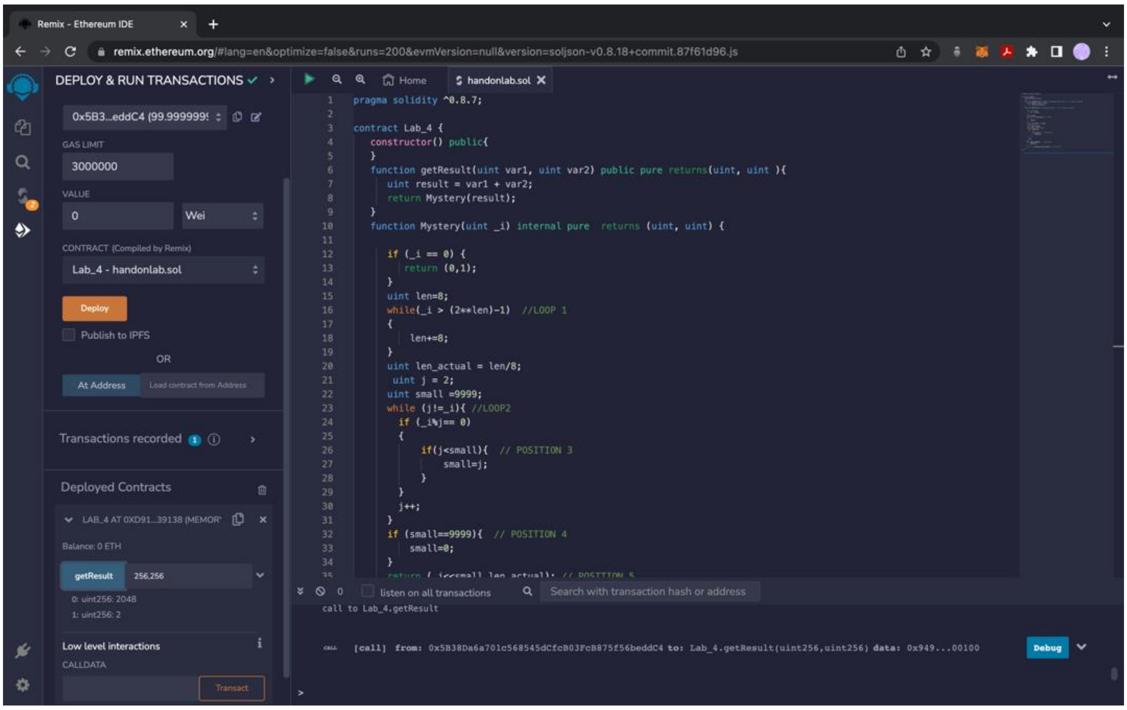
CALLDATA

Transact

```
pragma solidity ^0.8.7;

contract Lab_4 {
   constructor() public{
   }
   function getResult(uint var1, uint var2) public pure returns(uint, uint ){
      uint result = var1 + var2;
      return Mystery(result);
   }
   function Mystery(uint _i) internal pure  returns (uint, uint) {

      if (_i == 0) {
         return (0,1);
      }
      uint len=8;
      while(_i > (2**len)-1)  //LOOP 1
      {
         len+=8;
      }
      uint len_actual = len/8;
       uint j = 2;
      uint small =9999;
      while (j!=_i){ //LOOP2
        if (_i%j== 0)
        {
           if(j<small){  // POSITION 3
              small=j;
           }
        }
        j++;
      }
      if (small==9999){  // POSITION 4
         small=0;
      }
```

```
call to Lab_4.getResult

[call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: Lab_4.getResult(uint256,uint256) data: 0x949...00003
```

b. 256 and 256



DEPLOY & RUN TRANSACTIONS

0x5B3...eddC4 (99.999999

GAS LIMIT: 3000000

VALUE: 0 Wei

CONTRACT (Compiled by Remix): Lab_4 - handonlab.sol

Deploy

□ Publish to IPFS

OR

At Address | Load contract from Address

Transactions recorded 1

Deployed Contracts

LAB_4 AT 0XD91...39138 (MEMOR'

Balance: 0 ETH

getResult 256,256

0: uint256: 2048

1: uint256: 2

Low level interactions

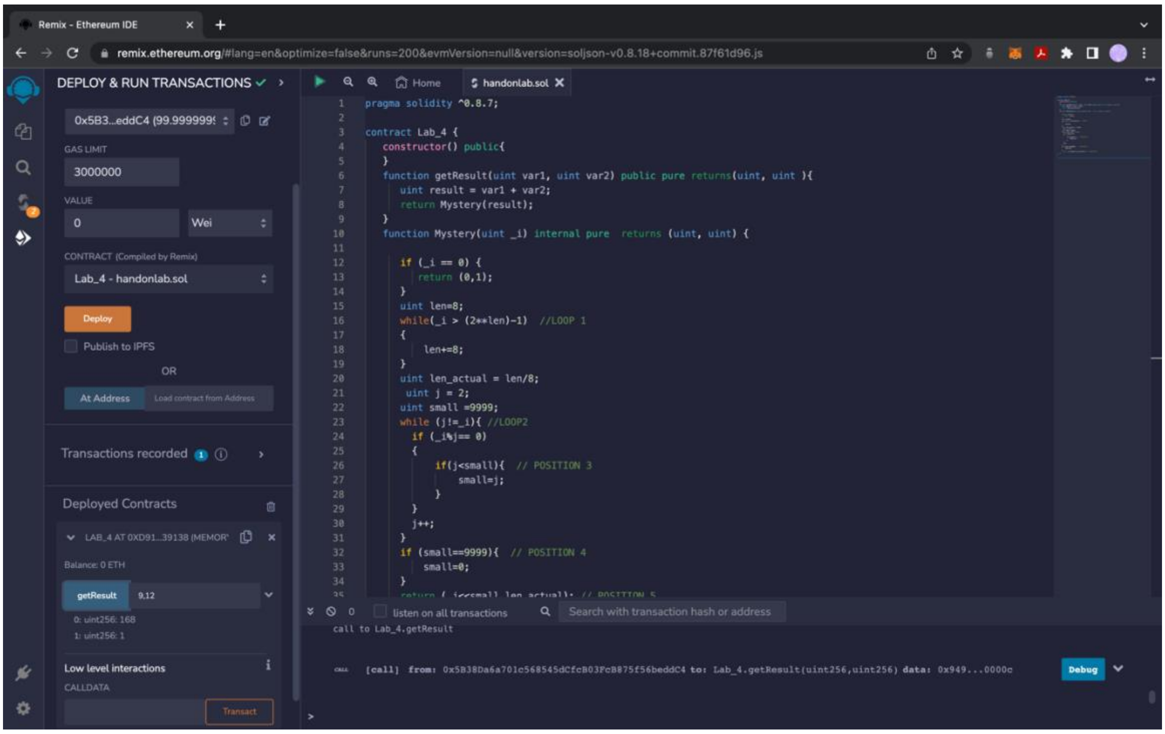
CALLDATA

Transact

```
pragma solidity ^0.8.7;

contract Lab_4 {
   constructor() public{
   }
   function getResult(uint var1, uint var2) public pure returns(uint, uint ){
      uint result = var1 + var2;
      return Mystery(result);
   }
   function Mystery(uint _i) internal pure  returns (uint, uint) {

      if (_i == 0) {
         return (0,1);
      }
      uint len=8;
      while(_i > (2**len)-1)  //LOOP 1
      {
         len+=8;
      }
      uint len_actual = len/8;
       uint j = 2;
      uint small =9999;
      while (j!=_i){ //LOOP2
        if (_i%j== 0)
        {
           if(j<small){  // POSITION 3
              small=j;
           }
        }
        j++;
      }
      if (small==9999){  // POSITION 4
         small=0;
      }
```

```
call to Lab_4.getResult

[call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: Lab_4.getResult(uint256,uint256) data: 0x949...00100
```

c. 9 and 12



Remix - Ethereum IDE
remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js
DEPLOY & RUN TRANSACTIONS
Home
handonlab.sol
Deploy
Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded 1
Deployed Contracts
Low level interactions
Transact
Debug
listen on all transactions
Search with transaction hash or address

ACCOUNT: 0x5B3...eddC4 (99.999999

GAS LIMIT: 3000000

VALUE: 0 Wei

CONTRACT (Compiled by Remix): Lab_4 - handonlab.sol

LAB_4 AT 0XD91...39138 (MEMOR'

Balance: 0 ETH

getResult 9,12

0: uint256: 168
1: uint256: 1

CALLDATA

```
pragma solidity ^0.8.7;

contract Lab_4 {
  constructor() public{
  }
  function getResult(uint var1, uint var2) public pure returns(uint, uint ){
    uint result = var1 + var2;
    return Mystery(result);
  }
  function Mystery(uint _i) internal pure  returns (uint, uint) {

    if (_i == 0) {
      return (0,1);
    }
    uint len=8;
    while(_i > (2**len)-1)  //LOOP 1
    {
      len+=8;
    }
    uint len_actual = len/8;
     uint j = 2;
    uint small =9999;
    while (j!=_i){ //LOOP2
      if (_i%j== 0)
      {
          if(j<small){  // POSITION 3
              small=j;
          }
      }
      j++;
    }
    if (small==9999){  // POSITION 4
      small=0;
    }
    return ( _i<<small, len_actual); // POSITION 5
```

```
call to Lab_4.getResult

[call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: Lab_4.getResult(uint256,uint256) data: 0x949...0000c
```

- State what the given program is doing by explaining the outputs generated in each test case.

For the test cases

1. 4, 3

The getResult function calculates the result value = 7 and inputs that into the Mystery function.

The Mystery function enters loop1 to set len to 8 and then goes to loop 2, where we find the smallest factor of 7. Since 7 is a prime number, small is set to 9999.

Thus, the final result is 7 and 1 since 7 is left shift by 0 bits and minimum number of bytes needed are 1.

2. 256, 256

The getResult function calculates the result value = 512 and inputs that into the Mystery function.

The Mystery function enters loop1 to set len to 16, since we need 2 bytes to represent 512 and then goes to loop 2, where we find the smallest factor of 512. The value of small is set to 2 (value of j) because 512 = 2^9.

Thus, the final result is 2048 and 2 since 512 is left shift by 2 bits and len_actual is 2.

3. 9,12

The getResult function calculates the result value = 21 and inputs that into the Mystery function.

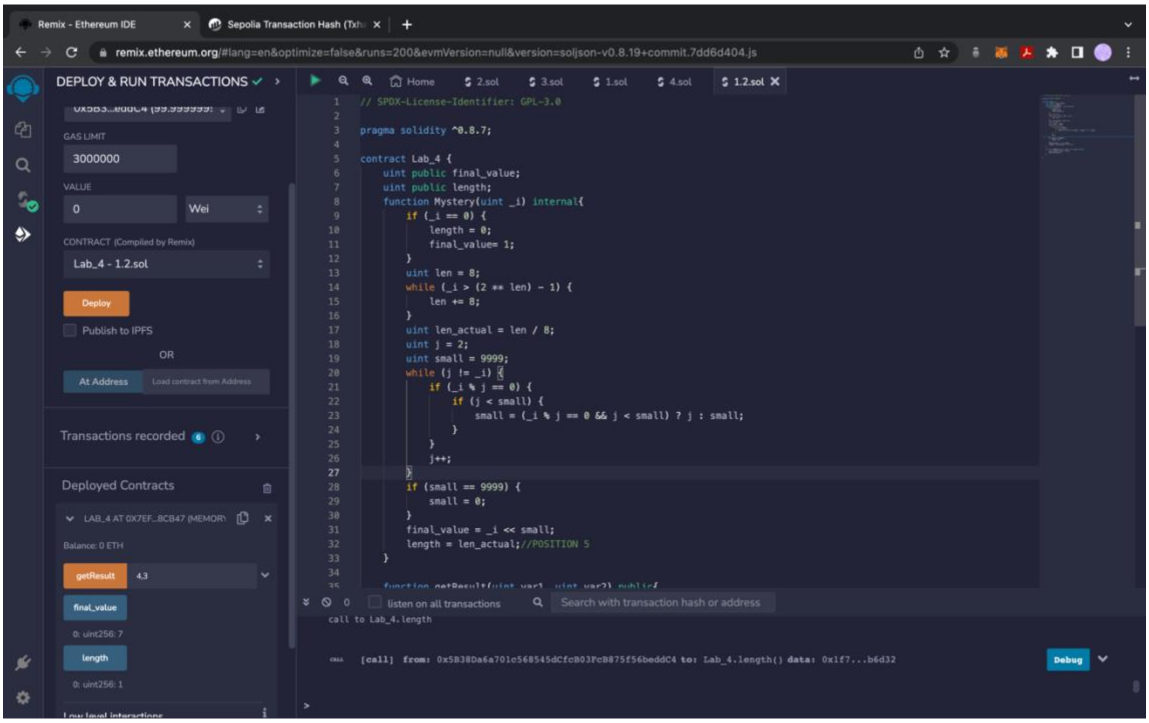
The Mystery function enters loop1 to set len to 8, since we need 1 byte to represent 21 and then goes to loop 2, where we find the smallest factor of 21. The value of small is set to 3 because 21 is divisible by 3.

Thus, the final result is 168 and 1 since 21 is left shift by 3 bits and len_actual is 1.

- Explain what **Loop1** and **Loop2** does in the given code.
  - Loop 1 checks the minimum number of bytes needed to represent the value of result. Loop 2 find the smallest factor of the value of result variable.
- Replace the statement in **Position 3** with a single line statement that does the same role/task as the statement(/s) given.
  - small = (j<small) ? j: small;

- What does **small** being assigned to 0 in the if condition at **Position 4** indicate?
    - This condition indicates that the value of result variable is a prime number, and that no smaller factor was found for it.
- Make appropriate changes to the **Mystery function** to return the two values without making use of a return statement. Attach a screenshot of the updated function (including the replacement line at Position 3).

```
// SPDX-License-Identifier: GPL-3.0
pragma solidity ^0.8.7;
contract Lab_4 {
  uint public final_value;
  uint public length;
  function Mystery(uint _i) internal{
    if (_i == 0) {
      length = 0;
      final_value= 1;
    }
    uint len = 8;
    while (_i > (2 ** len) - 1) {
      len += 8;
    }
    uint len_actual = len / 8;
    uint j = 2;
    uint small = 9999;
    while (j != _i) {
      if (_i % j == 0) {
        if (j < small) {
          small = (_i % j == 0 && j < small) ? j : small;
        }
      }
      j++;
    }
    if (small == 9999) {
      small = 0;
    }
    final_value = _i << small;
    length = len_actual;//POSITION 5
  }
  function getResult(uint var1, uint var2) public{
    uint result = var1 + var2;
    Mystery(result);
  }

}
```



Remix - Ethereum IDE
Sepolia Transaction Hash (Txh
remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.19+commit.7dd6d404.js
DEPLOY & RUN TRANSACTIONS
Home
2.sol
3.sol
1.sol
4.sol
1.2.sol
Deploy
Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded 6
Deployed Contracts
Debug
listen on all transactions
Search with transaction hash or address
Low level interactions

GAS LIMIT 3000000

VALUE 0 Wei

CONTRACT (Compiled by Remix) Lab_4 - 1.2.sol

LAB_4 AT 0X7EF...8CB47 (MEMORY)

Balance: 0 ETH

getResult 4,3

final_value

0: uint256: 7

length

0: uint256: 1

```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.7;

contract Lab_4 {
    uint public final_value;
    uint public length;
    function Mystery(uint _i) internal{
        if (_i == 0) {
            length = 0;
            final_value= 1;
        }
        uint len = 8;
        while (_i > (2 ** len) - 1) {
            len += 8;
        }
        uint len_actual = len / 8;
        uint j = 2;
        uint small = 9999;
        while (j != _i) {
            if (_i % j == 0) {
                if (j < small) {
                    small = (_i % j == 0 && j < small) ? j : small;
                }
            }
            j++;
        }
        if (small == 9999) {
            small = 0;
        }
        final_value = _i << small;
        length = len_actual;//POSITION 5
    }

    function getResult(uint var1, uint var2) public{
```

```
call to Lab_4.length

[call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: Lab_4.length() data: 0x1f7...b6d32
```

II. Write the Solidity code for a contract **Contract_XYZ** that follows the given outline-

Set_Method() function takes the size of the array (**a1**) and the multiplication factor(**a_m**) from the user. The function returns the multiplication table of a_m till size a1.

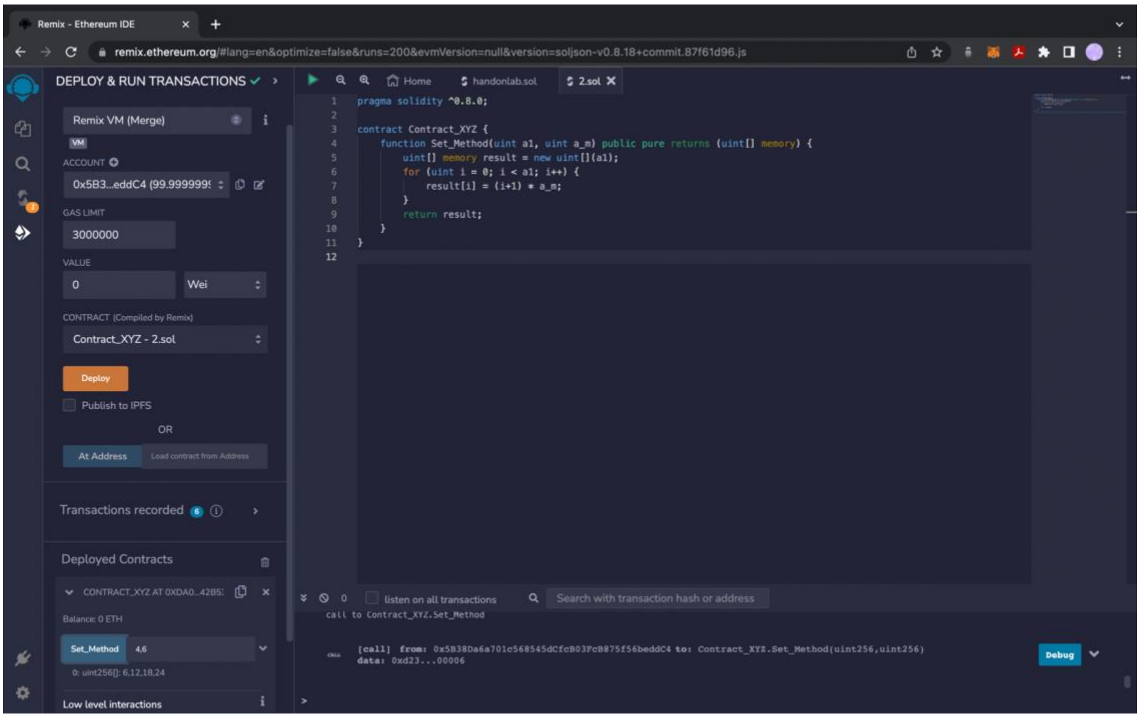
Example:

Input: 5(**a1**), 15(**a_m**)

Output: 0,15,30,45,60

Attach appropriate code and output screenshots in the document before submitting.

```
pragma solidity ^0.8.0;

contract Contract_XYZ {
   function Set_Method(uint a1, uint a_m) public pure returns (uint[] memory) {
      uint[] memory result = new uint[](a1);
      for (uint i = 0; i < a1; i++) {
         result[i] = (i+1) * a_m;
      }
      return result;
   }
}
```



Remix - Ethereum IDE
remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js
DEPLOY & RUN TRANSACTIONS
Home
handonlab.sol
2.sol
Deploy
At Address
Load contract from Address
Transactions recorded 6
Deployed Contracts
Low level interactions
Debug

Remix VM (Merge)

ACCOUNT: 0x5B3...eddC4 (99.999999!

GAS LIMIT: 3000000

VALUE: 0 Wei

CONTRACT (Compiled by Remix): Contract_XYZ - 2.sol

□ Publish to IPFS

CONTRACT_XYZ AT 0XDA0...42B5:

Balance: 0 ETH

Set_Method 4,6

0: uint256[]: 6,12,18,24

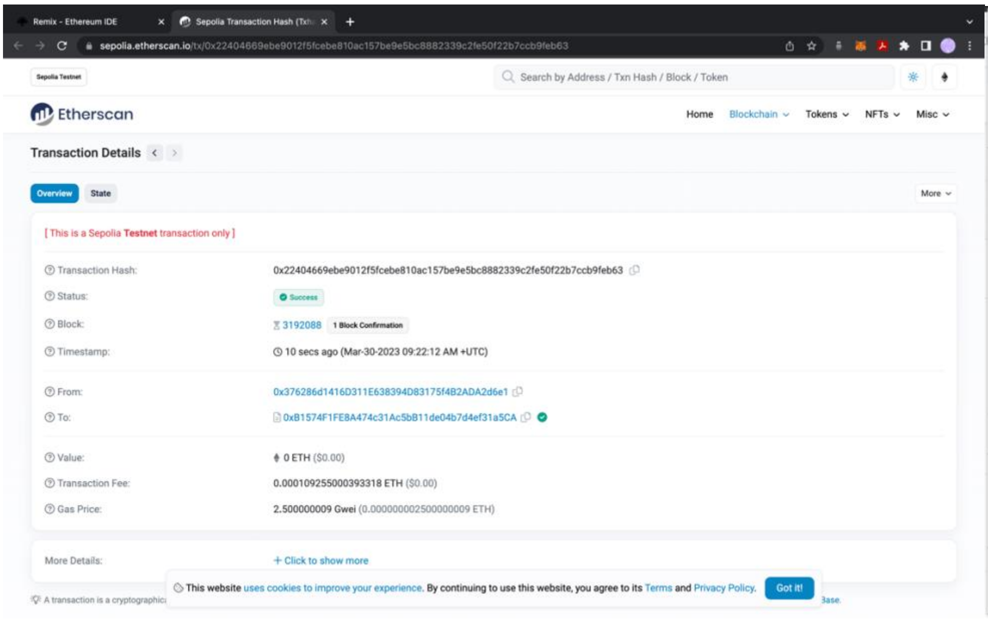
```
pragma solidity ^0.8.0;

contract Contract_XYZ {
    function Set_Method(uint a1, uint a_m) public pure returns (uint[] memory) {
        uint[] memory result = new uint[](a1);
        for (uint i = 0; i < a1; i++) {
            result[i] = (i+1) * a_m;
        }
        return result;
    }
}
```

```
call to Contract_XYZ.Set_Method

[call] from: 0x5B38Da6a701c568545dCfcB03FcB875f56beddC4 to: Contract_XYZ.Set_Method(uint256,uint256)
data: 0xd23...00006
```

III. Analyse the given codes on REMIX platform Metamask environment variable.

Take appropriate screenshots of the outputs and meta mask wallets after deploying the smart contracts.

Code 1



Remix - Ethereum IDE
Sepolia Transaction Hash (Txh
sepolia.etherscan.io/tx/0x22404669ebe9012f5fcebe810ac157be9e5bc8882339c2fe50f22b7ccb9feb63
Sepolia Testnet
Search by Address / Txn Hash / Block / Token
Etherscan
Home
Blockchain
Tokens
NFTs
Misc
Overview
State
More
Got it!

Transaction Details

[This is a Sepolia **Testnet** transaction only]

Transaction Hash: 0x22404669ebe9012f5fcebe810ac157be9e5bc8882339c2fe50f22b7ccb9feb63

Status: Success

Block: 3192088 1 Block Confirmation

Timestamp: 10 secs ago (Mar-30-2023 09:22:12 AM +UTC)

From: 0x376286d1416D311E638394D83175f4B2ADA2d6e1

To: 0xB1574F1FE8A474c31Ac5bB11de04b7d4ef31a5CA

Value: 0 ETH ($0.00)

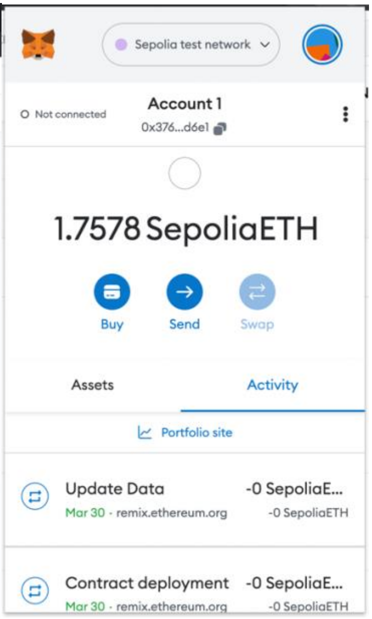
Transaction Fee: 0.000109255000393318 ETH ($0.00)

Gas Price: 2.500000009 Gwei (0.000000002500000009 ETH)

More Details: + Click to show more

A transaction is a cryptographic

This website uses cookies to improve your experience. By continuing to use this website, you agree to its Terms and Privacy Policy.



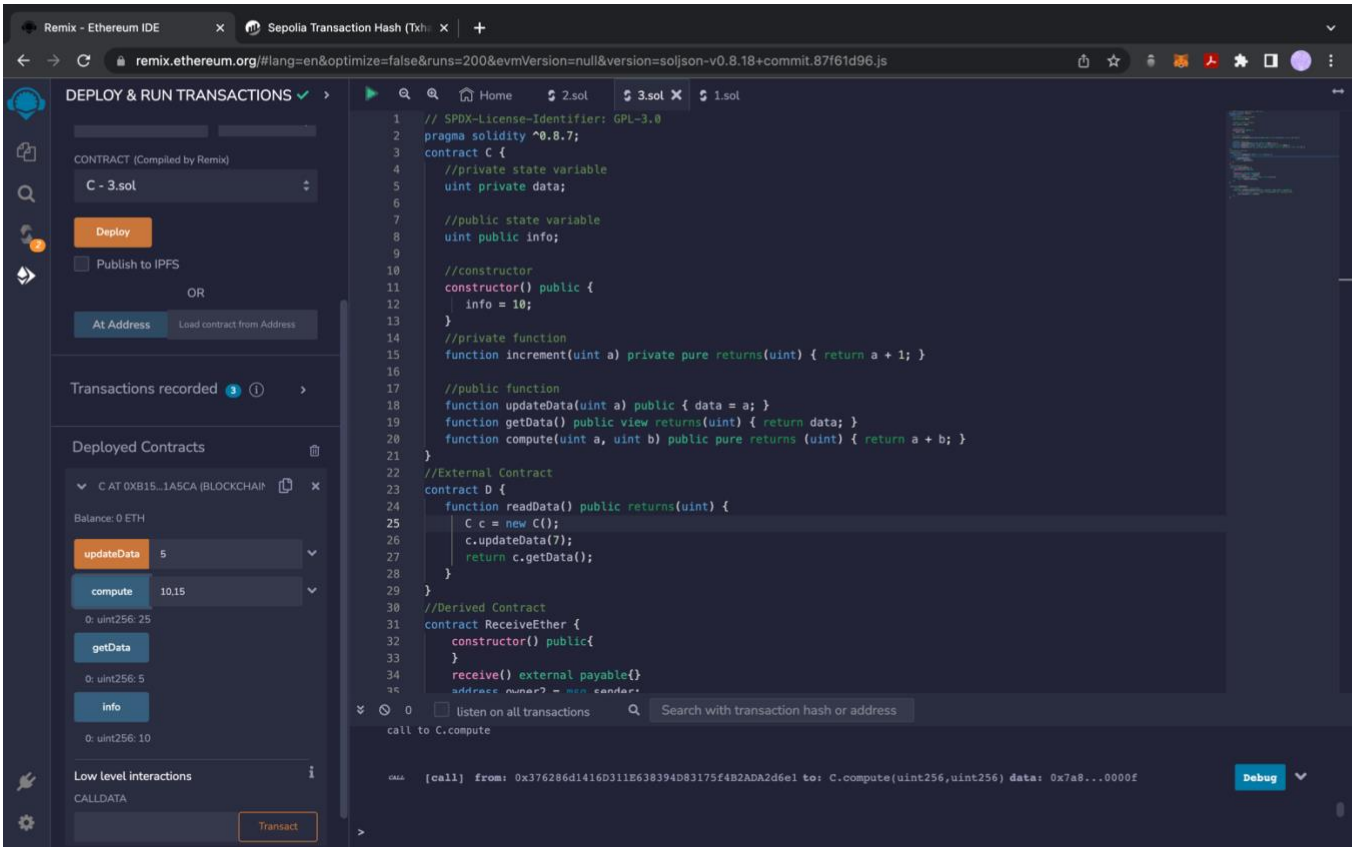
Account 1

0x376...d6e1

1.7578 SepoliaETH

Update Data -0 SepoliaE...
Mar 30 · remix.ethereum.org -0 SepoliaETH

Contract deployment -0 SepoliaE...
Mar 30 · remix.ethereum.org -0 SepoliaETH


Remix - Ethereum IDE
Sepolia Transaction Hash (Txh
remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js
DEPLOY & RUN TRANSACTIONS
Home
2.sol
3.sol
1.sol

CONTRACT (Compiled by Remix)

C - 3.sol

Deploy

□ Publish to IPFS

OR

At Address | Load contract from Address

Transactions recorded 3

Deployed Contracts

C AT 0XB15...1A5CA (BLOCKCHAIN

Balance: 0 ETH

updateData | 5

compute | 10,15

0: uint256: 25

getData

0: uint256: 5

info

0: uint256: 10

Low level interactions

CALLDATA

Transact

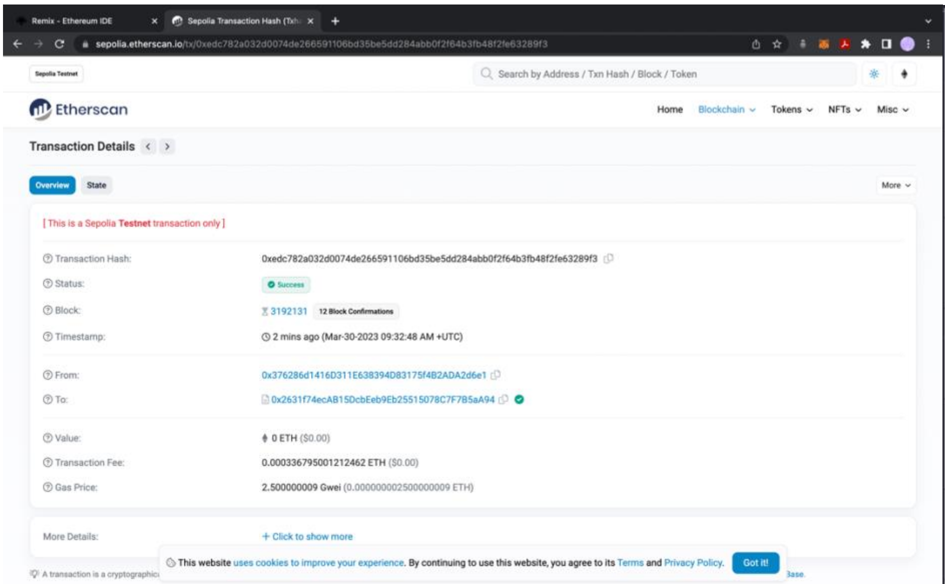
```
// SPDX-License-Identifier: GPL-3.0
pragma solidity ^0.8.7;
contract C {
   //private state variable
   uint private data;

   //public state variable
   uint public info;

   //constructor
   constructor() public {
      info = 10;
   }
   //private function
   function increment(uint a) private pure returns(uint) { return a + 1; }

   //public function
   function updateData(uint a) public { data = a; }
   function getData() public view returns(uint) { return data; }
   function compute(uint a, uint b) public pure returns (uint) { return a + b; }
}
//External Contract
contract D {
   function readData() public returns(uint) {
      C c = new C();
      c.updateData(7);
      return c.getData();
   }
}
//Derived Contract
contract ReceiveEther {
    constructor() public{
    }
    receive() external payable{}
    address owner2 = msg.sender;
```

□ listen on all transactions | Search with transaction hash or address

```
call to C.compute

CALL [call] from: 0x376286d1416D311E638394D83175f4B2ADA2d6e1 to: C.compute(uint256,uint256) data: 0x7a8...0000f
```

Debug

Code 2


Remix - Ethereum IDE
Sepolia Transaction Hash (Txh
sepolia.etherscan.io/tx/0xedc782a032d0074de266591106bd35be5dd284abb0f2f64b3fb48f2fe63289f3
Sepolia Testnet
Search by Address / Txn Hash / Block / Token
Etherscan
Home
Blockchain
Tokens
NFTs
Misc

Transaction Details

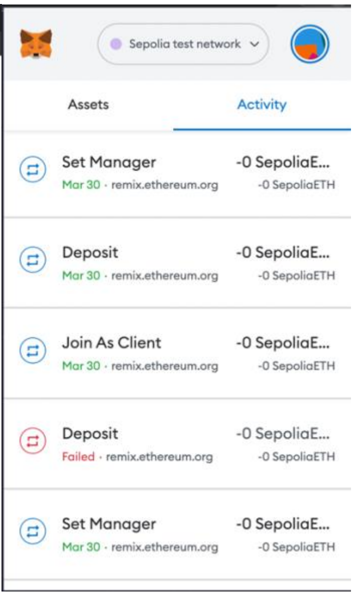
Overview | State | More

[This is a Sepolia **Testnet** transaction only]

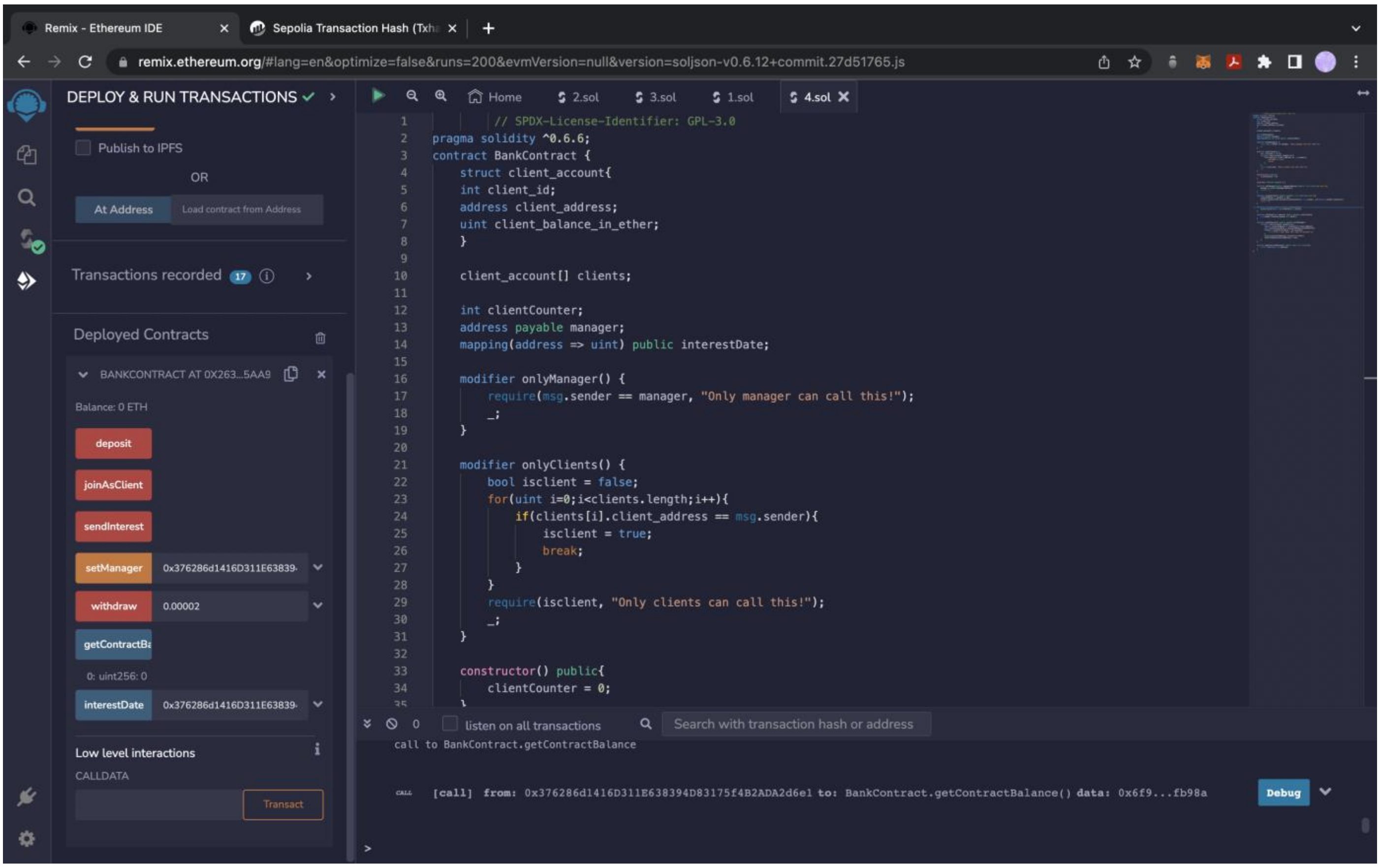
Transaction Hash:	0xedc782a032d0074de266591106bd35be5dd284abb0f2f64b3fb48f2fe63289f3
Status:	Success
Block:	3192131 12 Block Confirmations
Timestamp:	2 mins ago (Mar-30-2023 09:32:48 AM +UTC)
From:	0x376286d1416D311E638394D83175f4B2ADA2d6e1
To:	0x2631f74ecAB15DcbEeb9Eb25515078C7F7B5aA94
Value:	0 ETH ($0.00)
Transaction Fee:	0.000336795001212462 ETH ($0.00)
Gas Price:	2.500000009 Gwei (0.000000002500000009 ETH)
More Details:	+ Click to show more

This website uses cookies to improve your experience. By continuing to use this website, you agree to its Terms and Privacy Policy. Got it!

A transaction is a cryptographic



Set Manager Mar 30 · remix.ethereum.org	-0 SepoliaE... -0 SepoliaETH
Deposit Mar 30 · remix.ethereum.org	-0 SepoliaE... -0 SepoliaETH
Join As Client Mar 30 · remix.ethereum.org	-0 SepoliaE... -0 SepoliaETH
Deposit Failed · remix.ethereum.org	-0 SepoliaE... -0 SepoliaETH
Set Manager Mar 30 · remix.ethereum.org	-0 SepoliaE... -0 SepoliaETH



Remix - Ethereum IDE
Sepolia Transaction Hash (Txha
remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.6.12+commit.27d51765.js
DEPLOY & RUN TRANSACTIONS ✓
Home
2.sol
3.sol
1.sol
4.sol
□ Publish to IPFS
OR
At Address
Load contract from Address
Transactions recorded 17
Deployed Contracts
BANKCONTRACT AT 0X263...5AA9
deposit
joinAsClient
sendInterest
setManager
withdraw
getContractBa
interestDate
Low level interactions
CALLDATA
Transact
0
□ listen on all transactions
Search with transaction hash or address
Debug

```
    // SPDX-License-Identifier: GPL-3.0
pragma solidity ^0.6.6;
contract BankContract {
    struct client_account{
    int client_id;
    address client_address;
    uint client_balance_in_ether;
    }

    client_account[] clients;

    int clientCounter;
    address payable manager;
    mapping(address => uint) public interestDate;

    modifier onlyManager() {
        require(msg.sender == manager, "Only manager can call this!");
        _;
    }

    modifier onlyClients() {
        bool isclient = false;
        for(uint i=0;i<clients.length;i++){
            if(clients[i].client_address == msg.sender){
                isclient = true;
                break;
            }
        }
        require(isclient, "Only clients can call this!");
        _;
    }

    constructor() public{
        clientCounter = 0;
    }
```

Balance: 0 ETH

setManager: 0x376286d1416D311E63839

withdraw: 0.00002

getContractBa
0: uint256: 0

interestDate: 0x376286d1416D311E63839

```
call to BankContract.getContractBalance

CALL  [call] from: 0x376286d1416D311E638394D83175f4B2ADA2d6e1 to: BankContract.getContractBalance() data: 0x6f9...fb98a

>
```